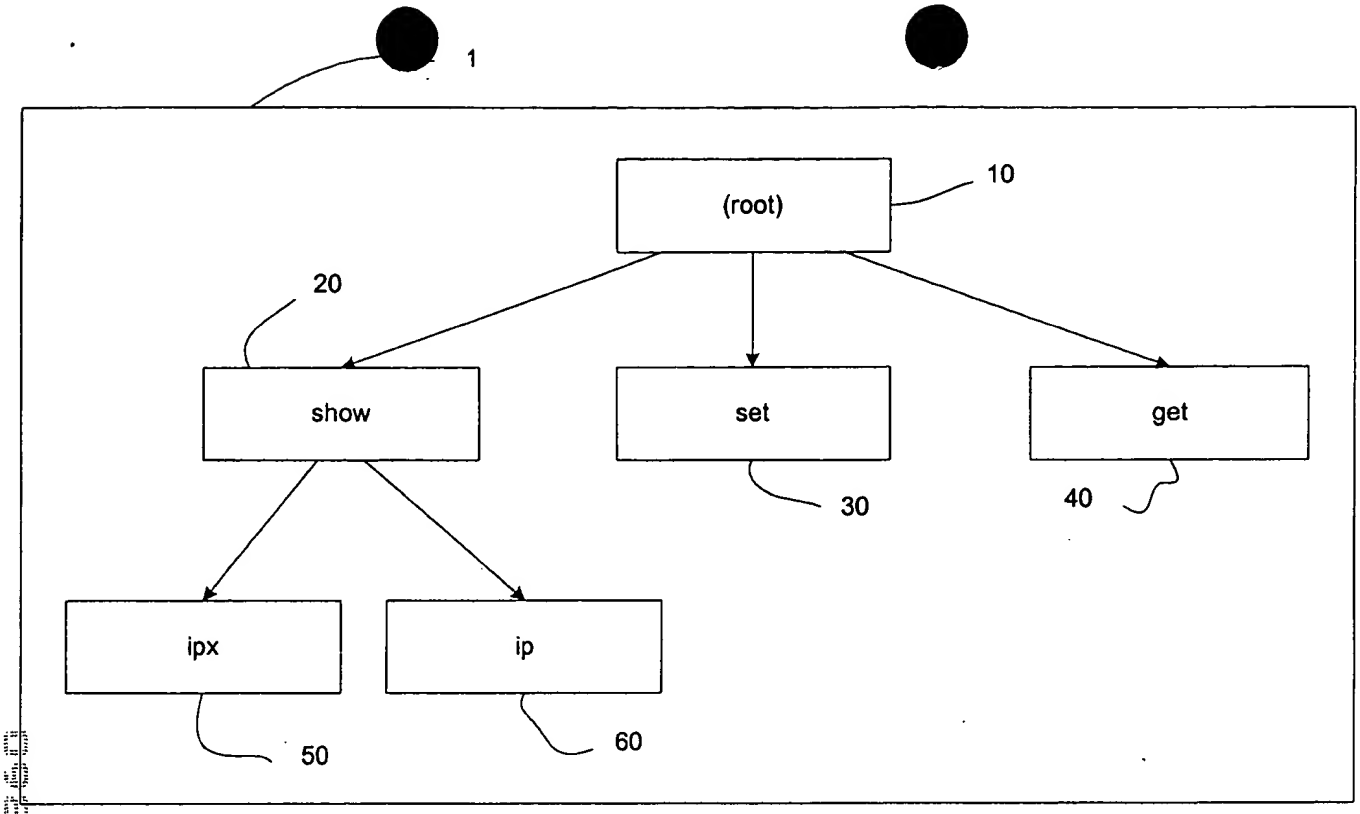

1
(root)
10
20
show
set
get
30
40
ipx
ip
50
60

Fig. 1

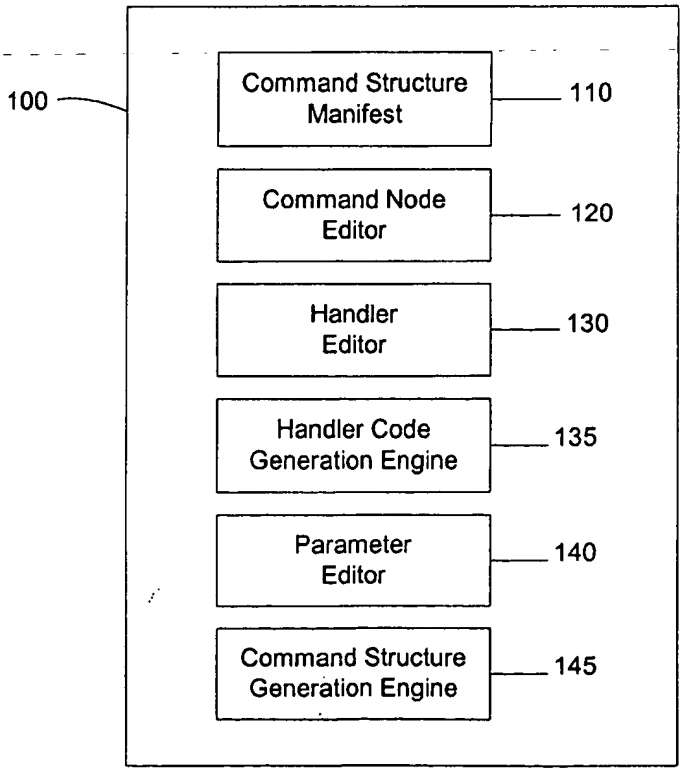

100
Command Structure Manifest
110
Command Node Editor
120
Handler Editor
130
Handler Code Generation Engine
135
Parameter Editor
140
Command Structure Generation Engine
145

Fig. 2

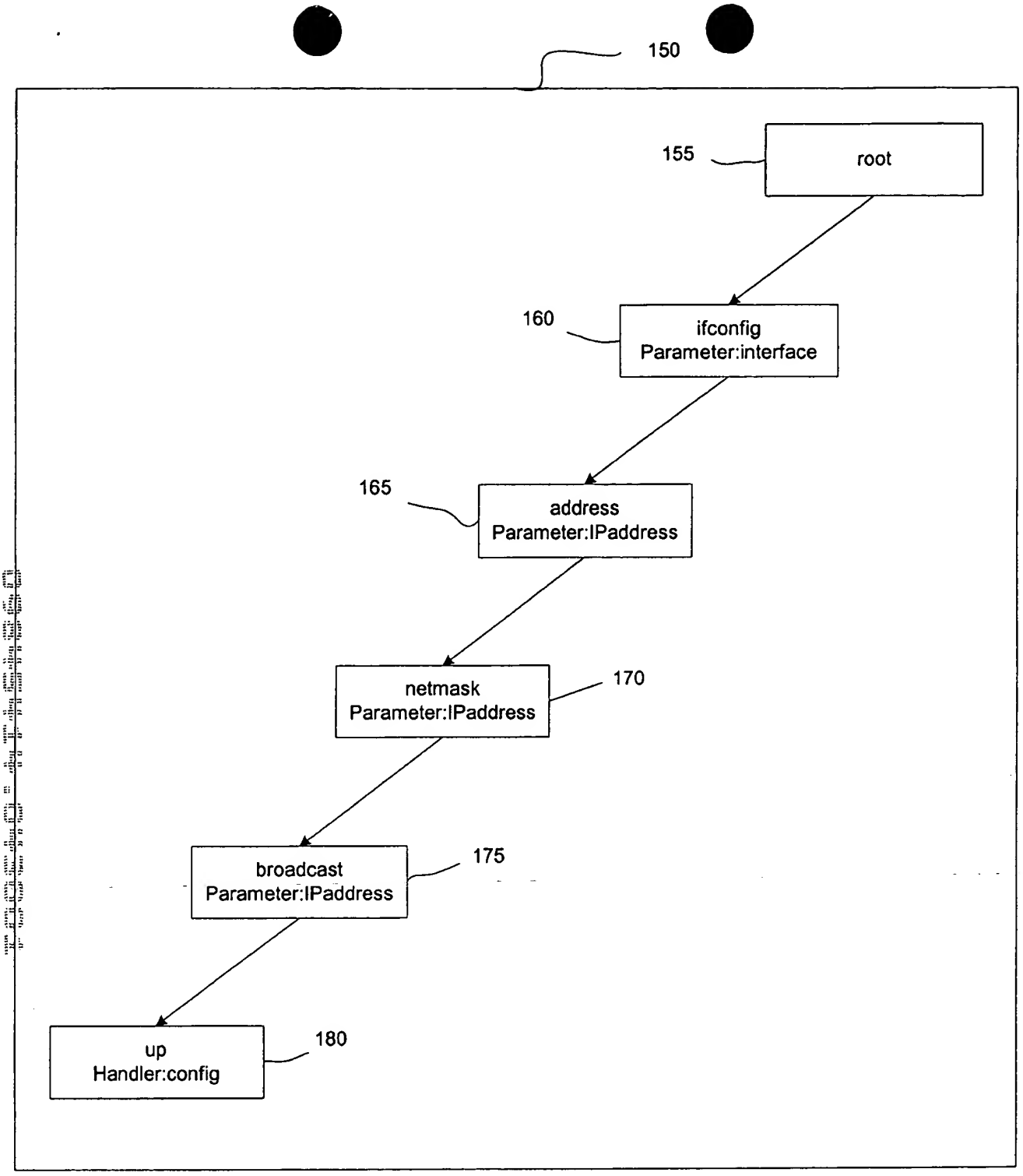

150
155
root
160
ifconfig
Parameter:interface
165
address
Parameter:IPaddress
netmask
Parameter:IPaddress
170
broadcast
Parameter:IPaddress
175
up
Handler:config
180

Fig. 3

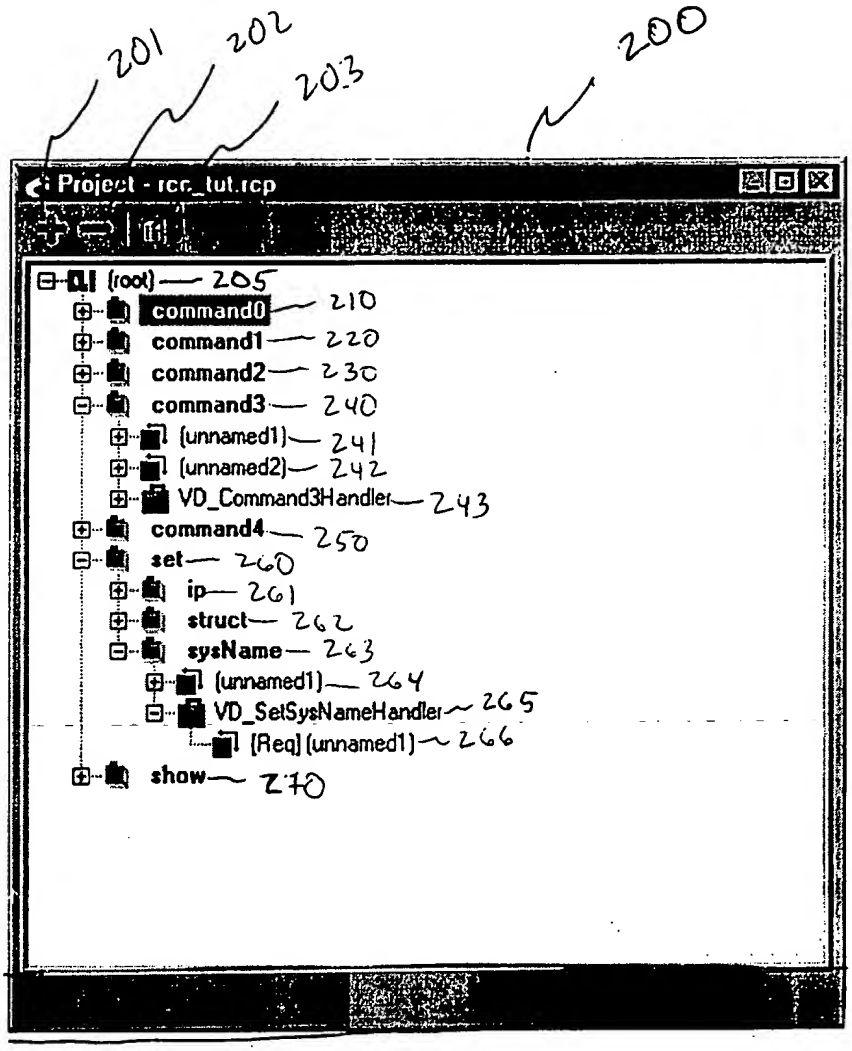

201
202
203
200
Project - rcc_tut.rcp
(root) 205
command0 210
command1 220
command2 230
command3 240
(unnamed1) 241
(unnamed2) 242
VD_Command3Handler 243
command4 250
set 260
ip 261
struct 262
sysName 263
(unnamed1) 264
VD_SetSysNameHandler 265
[Req] (unnamed1) 266
show 270

Fig. 4

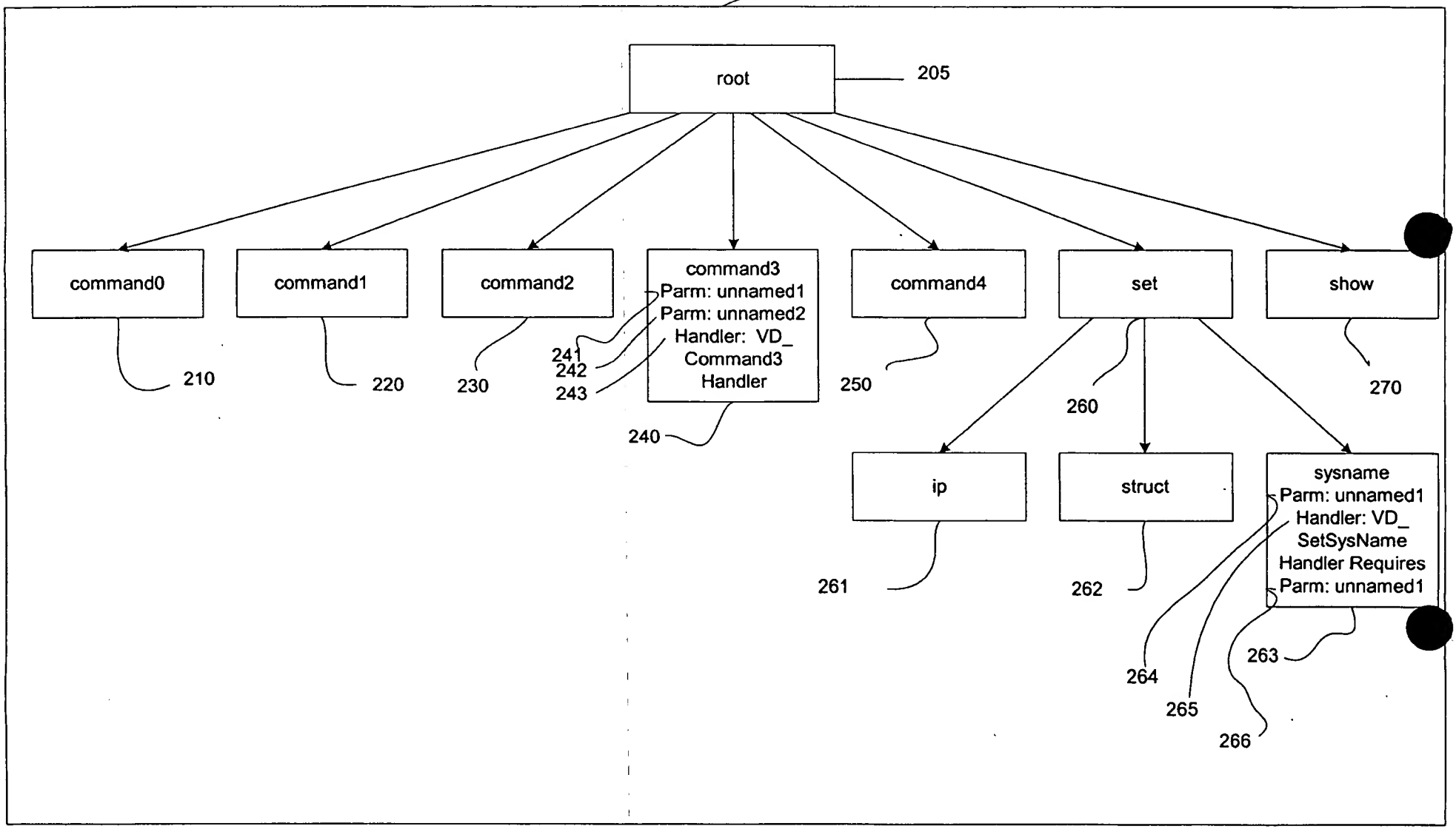

280
root
205
command0
command1
command2
command3
Parm: unnamed1
Parm: unnamed2
Handler: VD_
Command3
Handler
command4
set
show
210
220
230
241
242
243
240
250
260
270
ip
struct
sysname
Parm: unnamed1
Handler: VD_
SetSysName
Handler Requires
Parm: unnamed1
261
262
263
264
265
266

Fig. 5

Add a Command --300

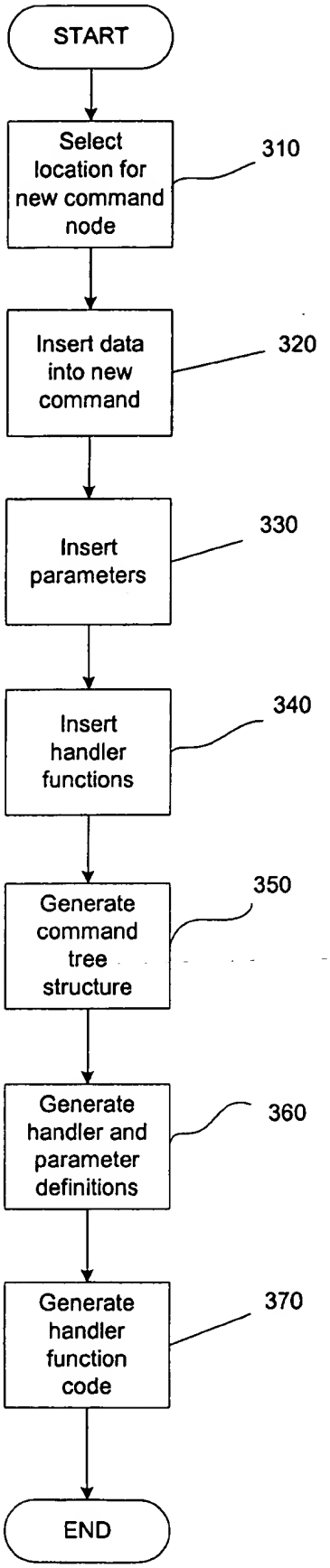


Fig. 6

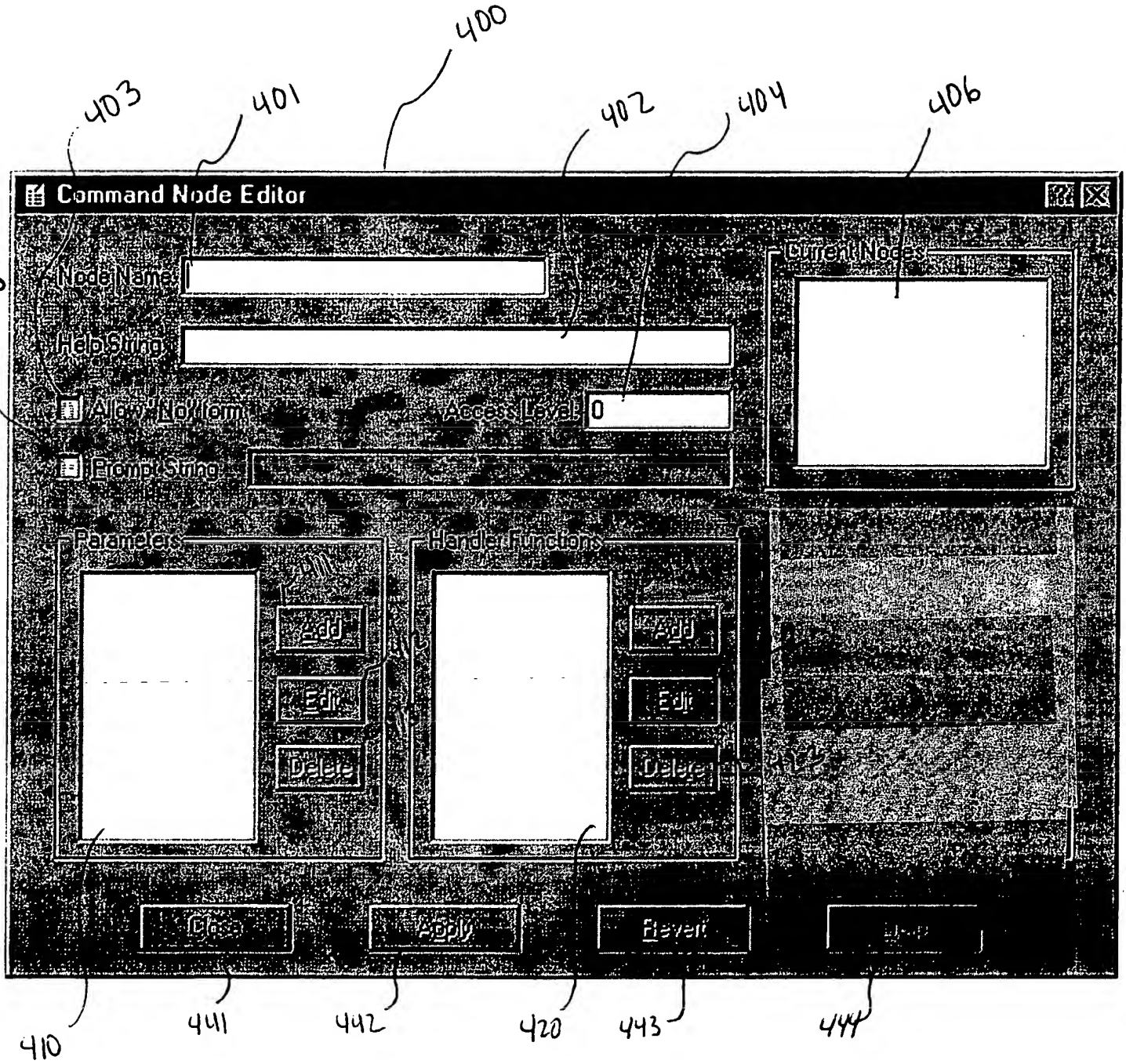

400
403
401
402
404
406
Command Node Editor
105
Node Name:
Help String:
Allow "No" Item
Access Level:
0
Prompt String:
Current Nodes
Parameters
Add
Edit
Delete
Handler Functions
Add
Edit
Delete
Close
Apply
Revert
410
441
442
420
443
444

Fig. 7

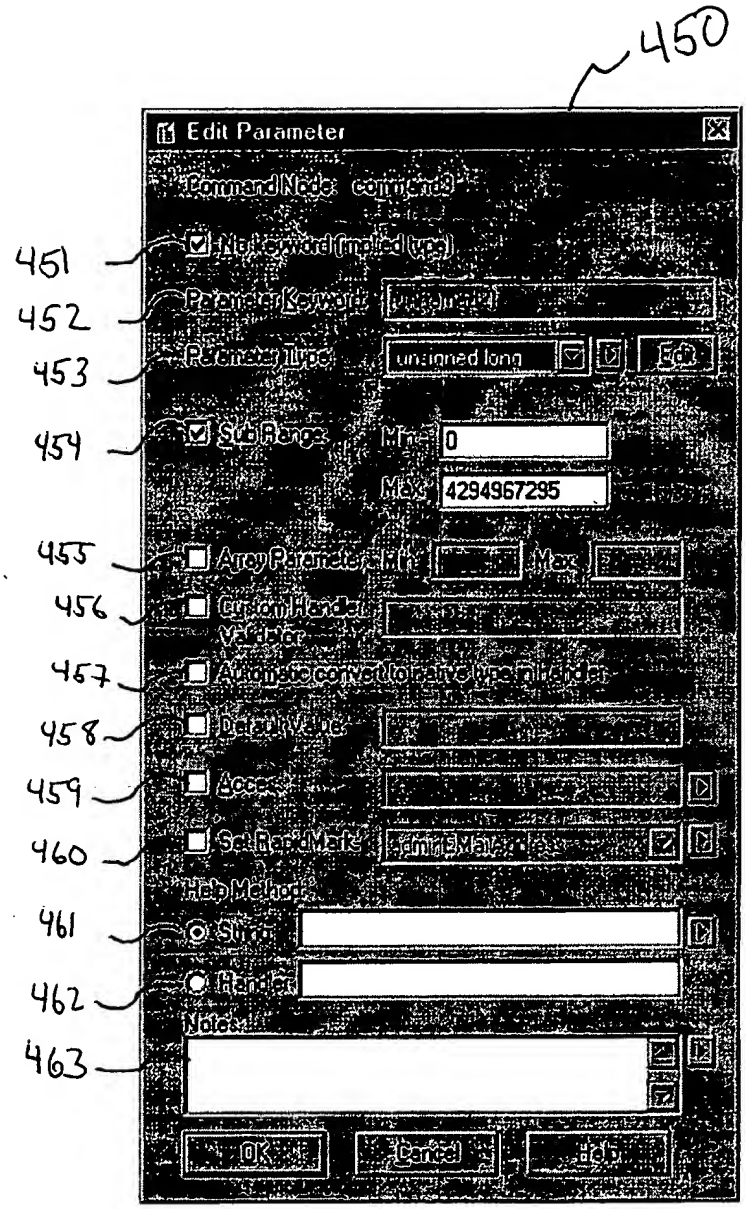

450
Edit Parameter
Command Node: command
451
452
453
454
455
456
457
458
459
460
461
462
463
Parameter Keyword:
Parameter Type:
unsigned long
Sub Range
Min
0
Max
4294967295
Array Parameter
Min
Max
Custom Handler
Variable
Default Value
Access
Set Back Mark
Help Methods
Stub
Handler
Notes:
OK
Cancel
Help

Fig. 8

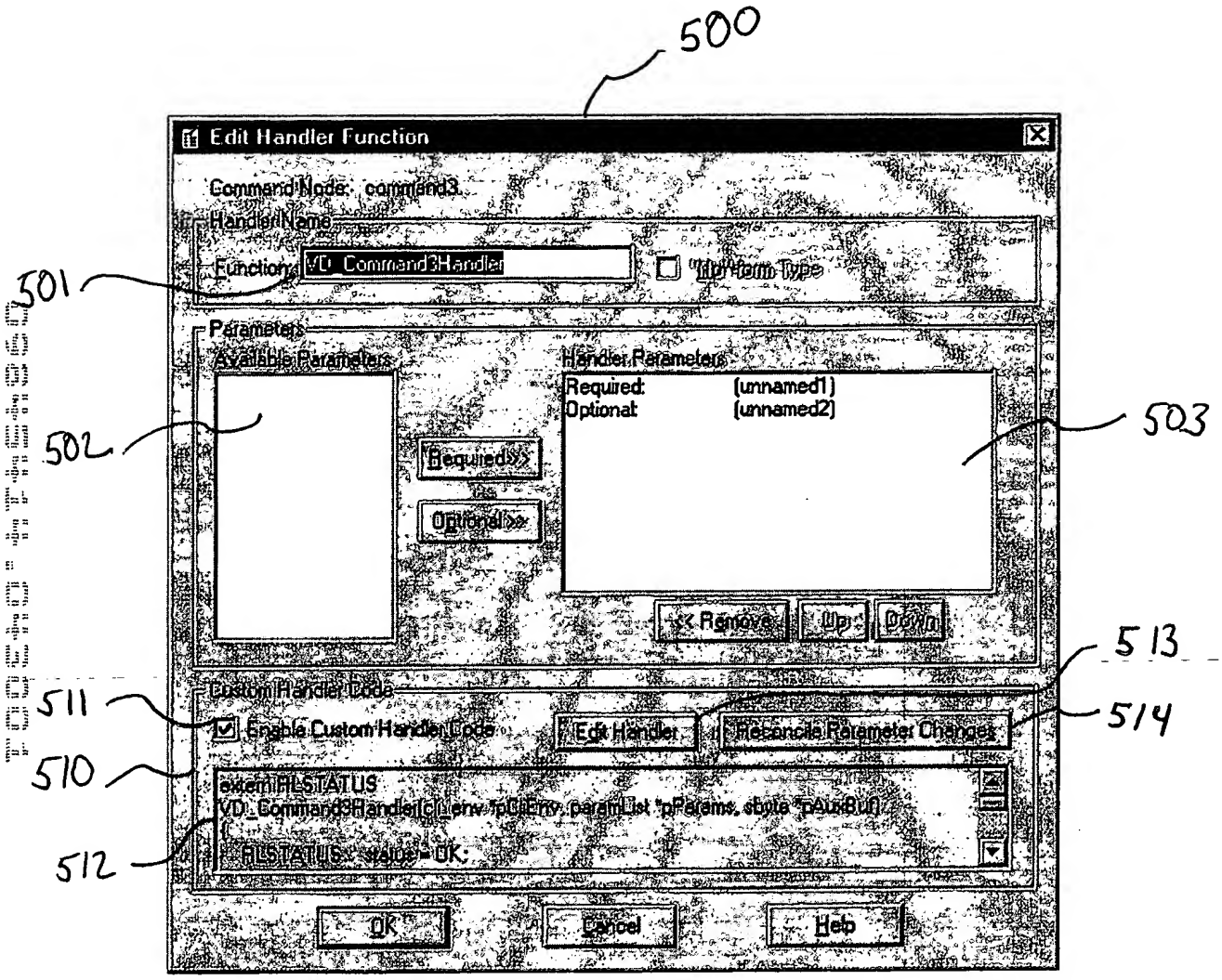

500
Edit Handler Function
Command Node: command3
Handler Name
Function: VD_Command3Handler
Uniform Type
Parameters
Available Parameters:
Handler Parameters:
Required: (unnamed1)
Optional: (unnamed2)
501
502
503
Required>>
Optional>>
<< Remove
Up
Down
513
511
514
Custom Handler Code
Enable Custom Handler Code
Edit Handler
Reconcile Parameter Changes
510
extern RLSTATUS
VD_Command3Handler(CL_env *pCLEnv, paramList *pParams, sbyte *pAux8Ur)
{
RLSTATUS status = OK;
512
OK
Cancel
Help

Fig. 9

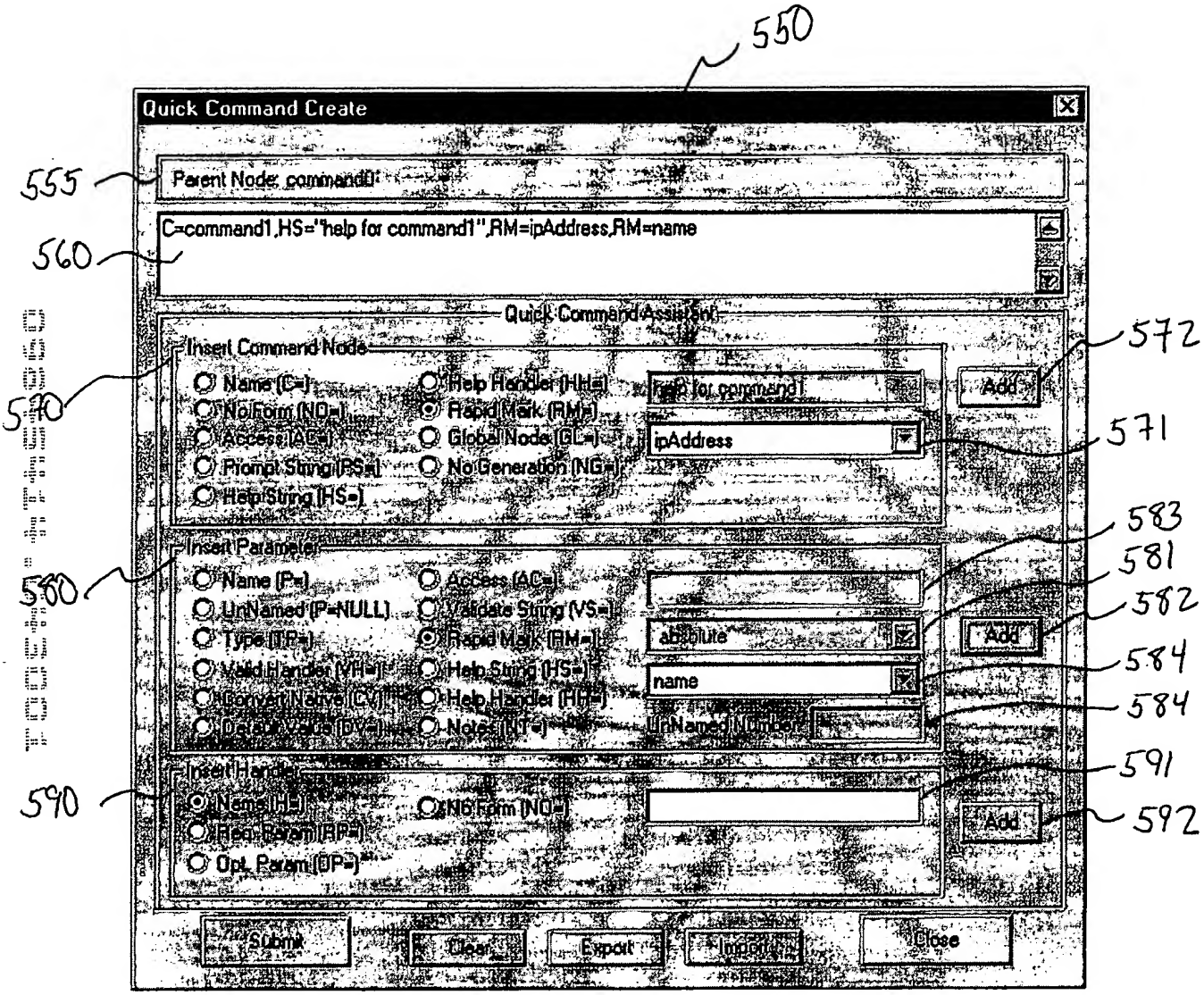

550
Quick Command Create
555
Parent Node: command1
C=command1,HS="help for command1",RM=ipAddress,RM=name
560
Quick Command Assistant
Insert Command Node
570
Name (C=)
No Form (NO=)
Access (AC=)
Prompt String (PS=)
Help String (HS=)
Help Handler (HH=)
Rapid Mark (RM=)
Global Node (GL=)
No Generation (NG=)
help for command1
ipAddress
Add
572
571
Insert Parameter
580
Name (P=)
UnNamed (P=NULL)
Type (TP=)
Valid Handler (VH=)
Convert Native (CV=)
Default Value (DV=)
Access (AC=)
Validate String (VS=)
Rapid Mark (RM=)
Help String (HS=)
Help Handler (HH=)
Notes (NT=)
absolute
name
UnNamed Number
Add
583
581
582
584
584
Insert Handler
590
Name (H=)
Req. Param (RP=)
Opt. Param (OP=)
No Form (NO=)
Add
591
592
Submit
Clear
Export
Import
Close

Fig. 10

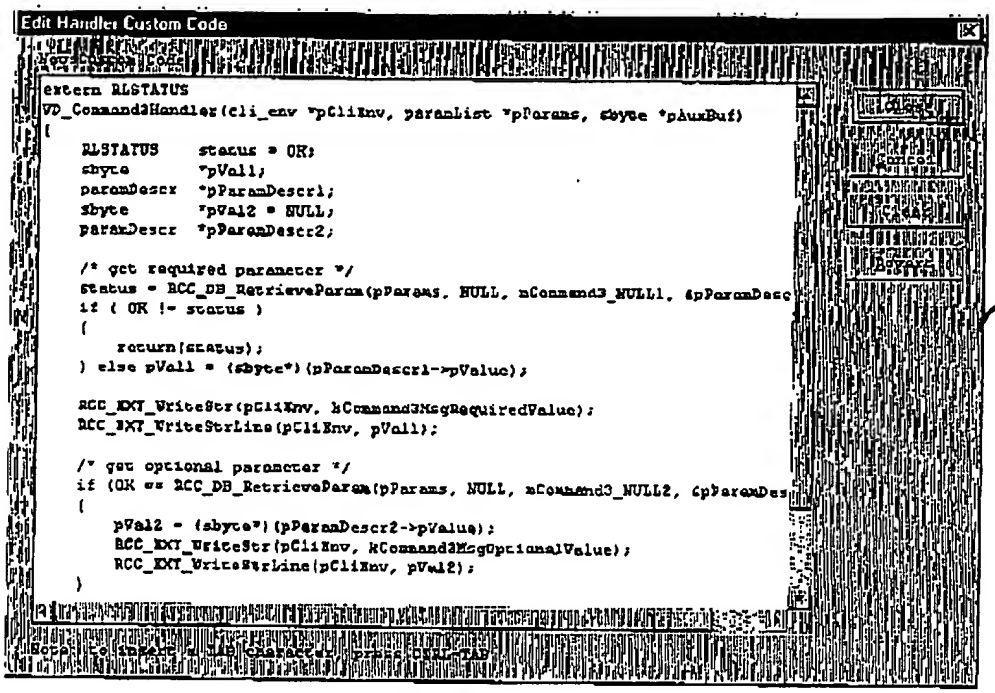

Edit Handler Custom Code
extern RLSTATUS
VD_Command3Handler(cli_env *pCliEnv, paramList *pParams, sbyte *pAuxBuf)
{
RLSTATUS status = OK;
sbyte *pVal1;
paramDescr *pParamDescr1;
sbyte *pVal2 = NULL;
paramDescr *pParamDescr2;
/* get required parameter */
status = RCC_DB_RetrieveParam(pParams, NULL, mCommand3_NULL1, &pParamDesc
if (OK != status)
{
return(status);
} else pVal1 = (sbyte*)(pParamDescr1->pValue);
RCC_EXT_WriteStr(pCliEnv, mCommand3MsgRequiredValue);
RCC_EXT_WriteStrLine(pCliEnv, pVal1);
/* get optional parameter */
if (OK == RCC_DB_RetrieveParam(pParams, NULL, mCommand3_NULL2, &pParamDes
{
pVal2 = (sbyte*)(pParamDescr2->pValue);
RCC_EXT_WriteStr(pCliEnv, mCommand3MsgOptionalValue);
RCC_EXT_WriteStrLine(pCliEnv, pVal2);
}
600

Fig. 11

Fig. 12

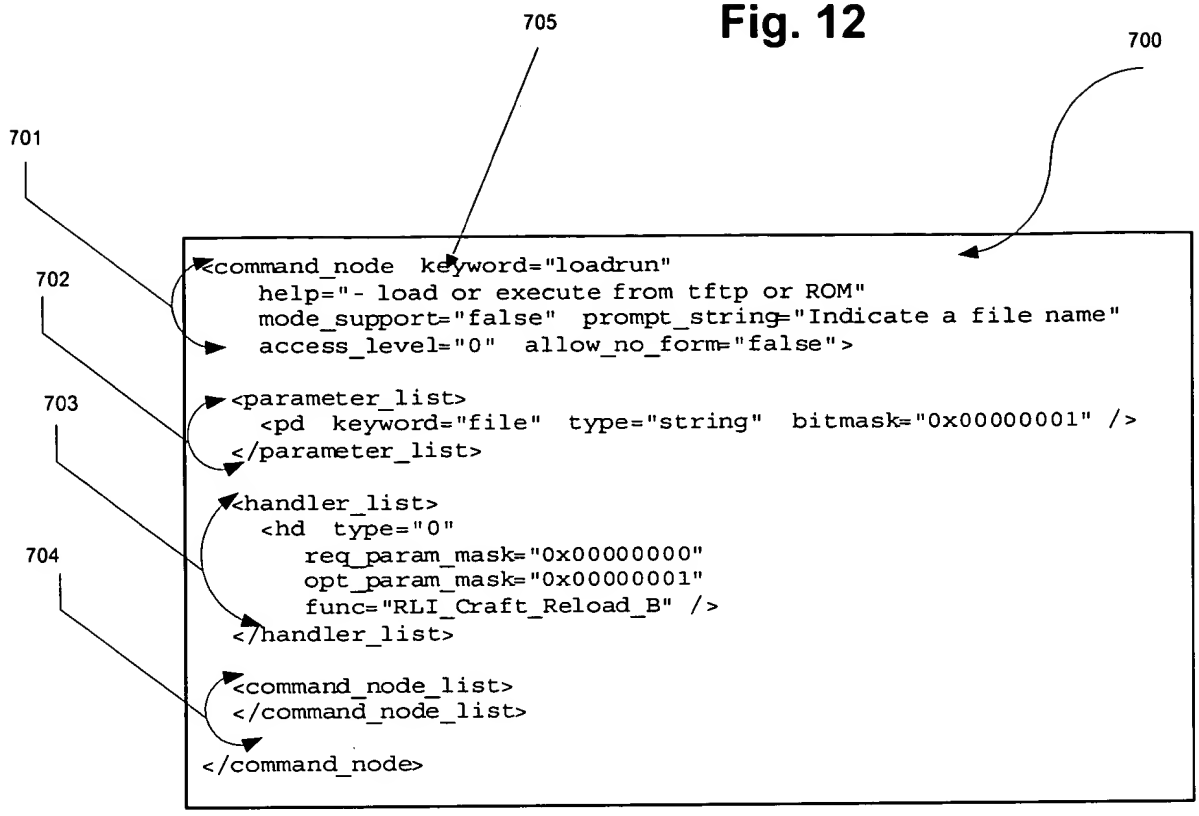

705
700
701
702
703
704
<command_node keyword="loadrun"
help="- load or execute from tftp or ROM"
mode_support="false" prompt_string="Indicate a file name"
access_level="0" allow_no_form="false">
<parameter_list>
<pd keyword="file" type="string" bitmask="0x00000001" />
</parameter_list>
<handler_list>
<hd type="0"
req_param_mask="0x00000000"
opt_param_mask="0x00000001"
func="RLI_Craft_Reload_B" />
</handler_list>
<command_node_list>
</command_node_list>
</command_node>

Fig. 13

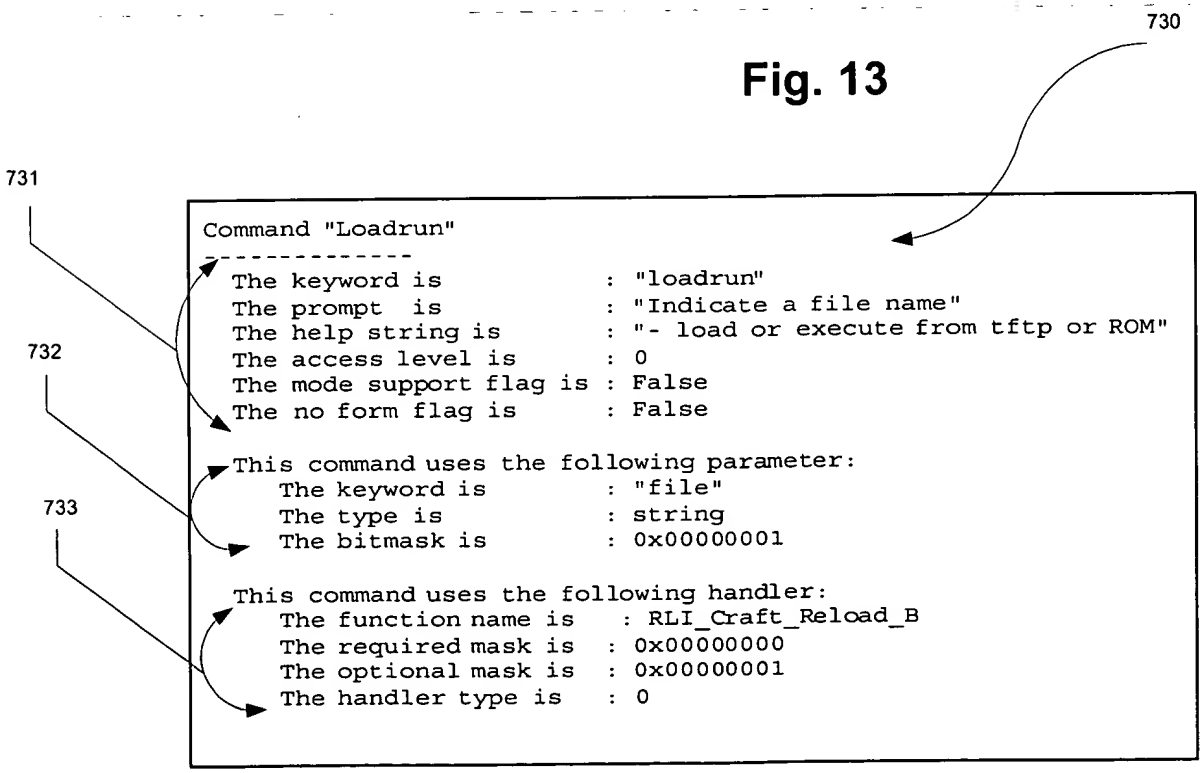
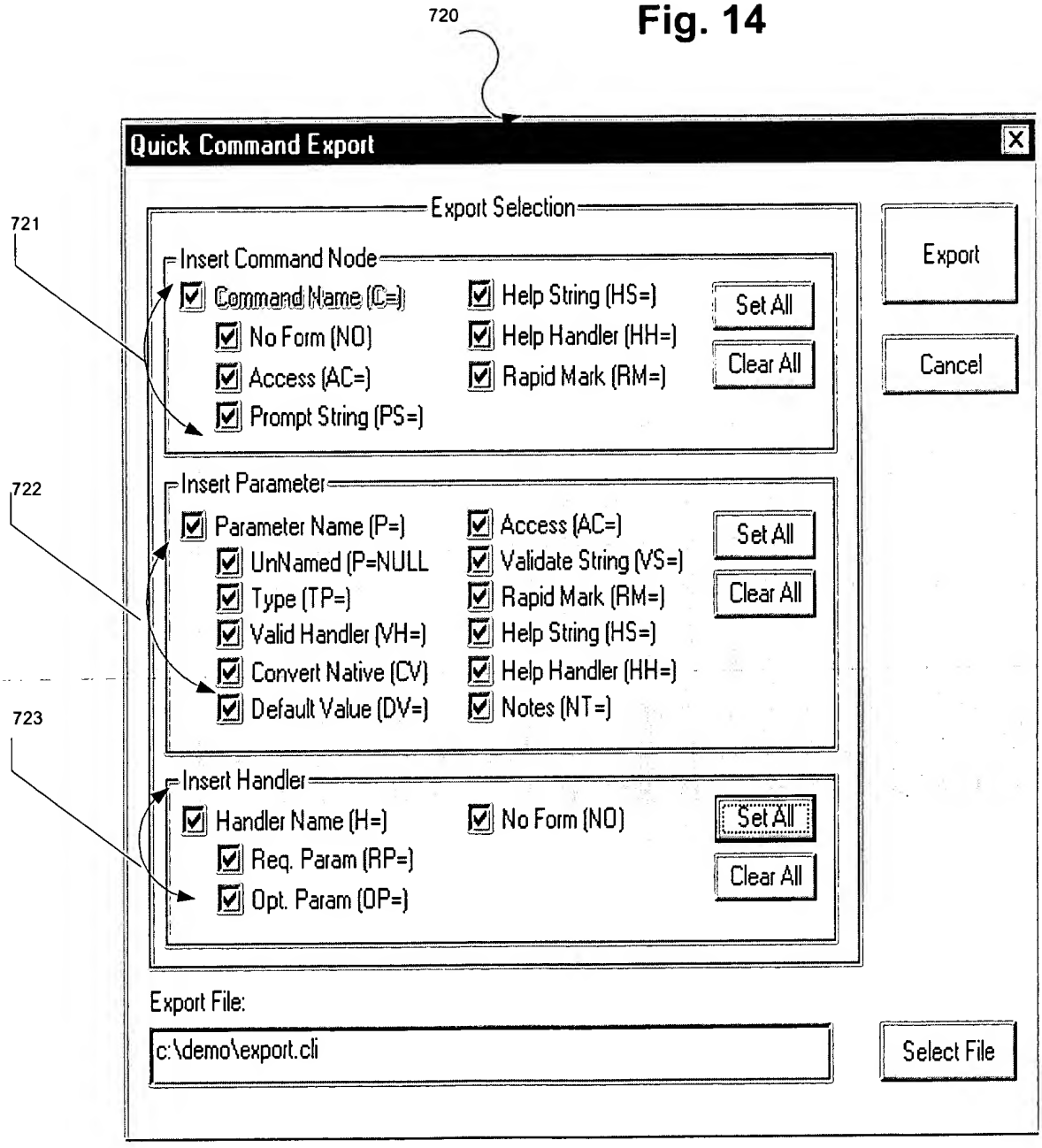

730
731
732
733
Command "Loadrun"
The keyword is : "loadrun"
The prompt is : "Indicate a file name"
The help string is : "- load or execute from tftp or ROM"
The access level is : 0
The mode support flag is : False
The no form flag is : False
This command uses the following parameter:
The keyword is : "file"
The type is : string
The bitmask is : 0x00000001
This command uses the following handler:
The function name is : RLI_Craft_Reload_B
The required mask is : 0x00000000
The optional mask is : 0x00000001
The handler type is : 0

Fig. 14


720
Quick Command Export
Export Selection
721
Insert Command Node
Command Name (C=)
No Form (NO)
Access (AC=)
Prompt String (PS=)
Help String (HS=)
Help Handler (HH=)
Rapid Mark (RM=)
Set All
Clear All
Export
Cancel
722
Insert Parameter
Parameter Name (P=)
UnNamed (P=NULL
Type (TP=)
Valid Handler (VH=)
Convert Native (CV)
Default Value (DV=)
Access (AC=)
Validate String (VS=)
Rapid Mark (RM=)
Help String (HS=)
Help Handler (HH=)
Notes (NT=)
Set All
Clear All
723
Insert Handler
Handler Name (H=)
Req. Param (RP=)
Opt. Param (OP=)
No Form (NO)
Set All
Clear All
Export File:
c:\demo\export.cli
Select File

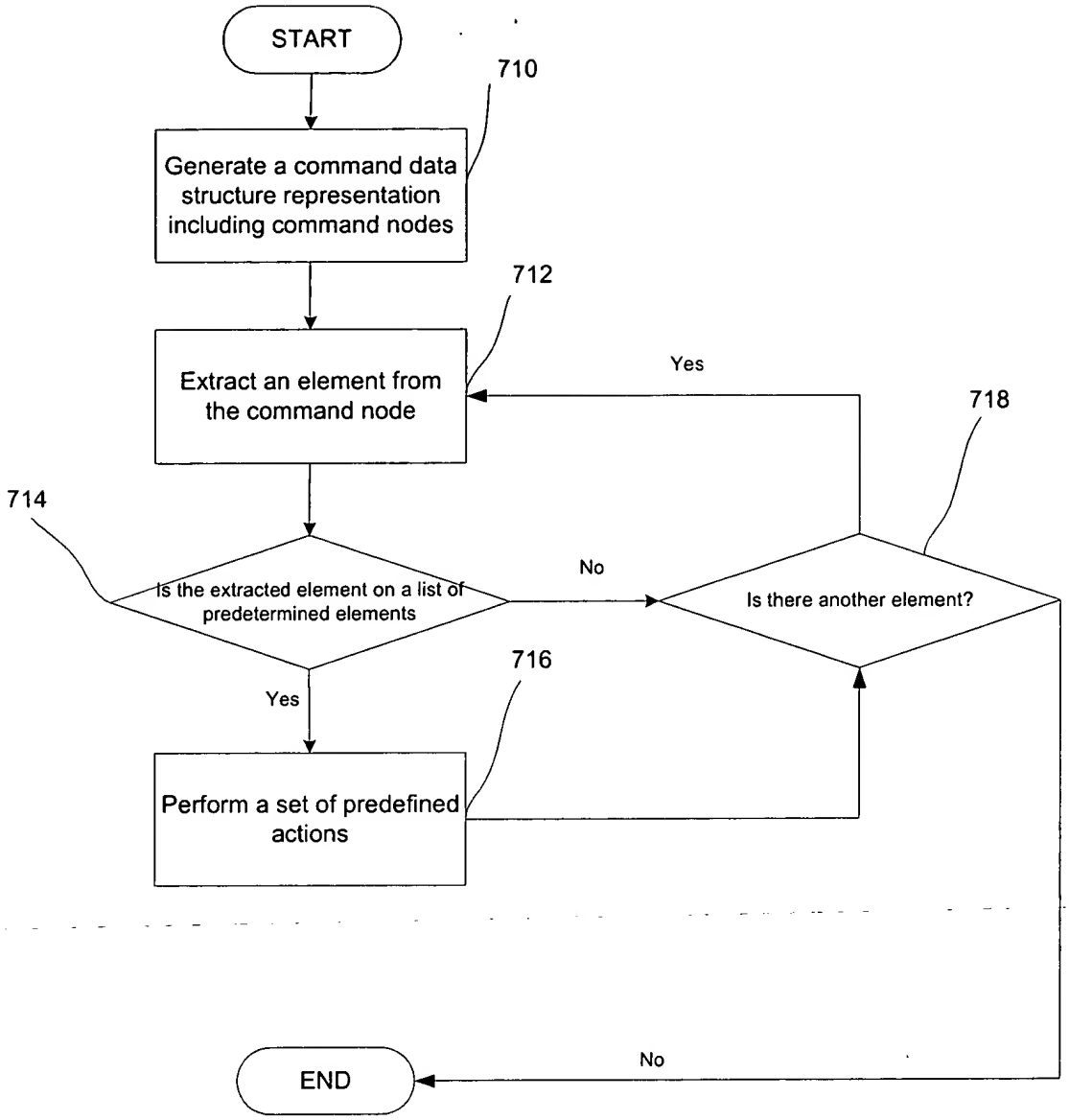

START
710
Generate a command data structure representation including command nodes
712
Extract an element from the command node
Yes
718
714
Is the extracted element on a list of predetermined elements
No
Is there another element?
Yes
716
Perform a set of predefined actions
No
END

Fig. 15